CONTENT-BASED IMAGE RETRIEVAL LOOKUP IN P2P DHT-BASE SYSTEM

IYAS ABDULLAH ,[a] MOHAMMAD AlODAT [b]
[a] University of Craiova, Romania
eyasao@yahoo.com
[b] University of Craiova, Romania
Systemout7@yahoo.com

Abstract. In P2P information discovery system that supports flexible queries using partial keywords and wild-cards, and range of queries. Users looking up resources stored in peer-to-peer systems have only a little information for identifying of resources. In This paper we describe techniques for look up specific image store in the DHT-base network to give it to the users.

Keywords: peer to peer, overlay systems, dht-based system.

1 INTRODUCTION

In the Peer-to-Peer (P2P) computing paradigm, entities at the edges of the network can directly interact as equals (or peers) and share information, services and resources without centralized servers. Key characteristics of these systems include decentralization, self-organization, dynamism and fault-tolerance, which make them naturally scalable and attractive solutions for applications. Similarly grid computing is rapidly emerging as the dominant paradigm for wide area distributed computing.

The DHT paradigms (Chord [10], Pastry [2], Tapestry [4] and CAN [9]) are appropriate for building large-scale distributed applications due to their scalability, fault-tolerance and self-organization. However, these DHTs are designed for exact key lookup. Range queries cannot be efficiently supported since consistent hashing mechanisms destroy data locality (nearby data points in the multi-dimensional data space are mapped to the same node or to nodes that are close together in the overlay network.

A major limitation of P2P DHT systems is that they only support exact-match lookups: one needs to know the exact key (identifier) of a data item to locate the nodes responsible for storing that item. Propose to augment P2P DHT systems with mechanisms for locating data using incomplete information. Note that we do not aim at answering complex database-like queries, but rather at providing practical techniques for searching data in a DHT.Indexing techniques can be layered on top of an arbitrary P2P DHT infrastructure, and thus benefit from any advanced features implemented in the DHT (e.g., replication, load-balancing). We have conducted a comprehensive evaluation that demonstrates their effectiveness in realistic settings.

For search by techniques content-based image retrieval (CBIR) "Content-based" means that the search will analyze the actual contents of the image rather than keywords, tags or descriptions associated with the image. The term 'content' in this context refers to colors, shapes, textures, or any other information that can be derived from the image itself. The earliest use the term content-based image as figure 1.

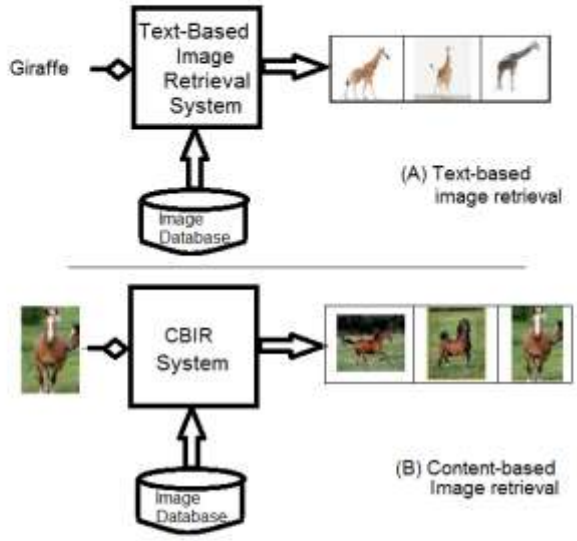

Figure 1: Image retrieval approaches

The text-based approaches associate keywords with each stored image in the database. These keywords are typically generated manually Figure 1a. And content-based image retrieval (CBIR), where retrieval of images is guided by providing a query image or a sketch generated by a user as Figure 1b.

Feature extraction is the basis of content-based image retrieval. CBIR operates on a principle of retrieving stored images from a database by comparing features automatically extracted from the images themselves. The commonest features used are mathematical measures of color, texture or shape.

Range partitioning designates each node responsible for one contiguous range of attribute values , and thus provides good locality. The amount of meta-data is fairly small, requiring just the attribute values at the partition boundaries. However, ensuring load balance across partitions as data evolves is a non-trivial problem.

The architecture of the presented P2P information retrieval system is similar to data-lookup systems [9, 10], Existing information storage/discovery systems can be broadly classified as unstructured or structured. Unstructured systems (such as Gnutella [5]) are based on flooding techniques and process queries by forwarding them to neighboring peers. While unstructured systems are relatively easy to maintain and can support complex queries, they do not guarantee that all matches to a query in a practical-sized system will be found. Furthermore, overheads of flooding can be significant, and a number of techniques have been proposed to reduce these overheads [4, 8]. Structured data lookup systems (e.g. CAN [9], Chord [10]) use structured overlay networks and consistent hashing to implement Internet-scale distributed hash tables.

We can see to at [3] Partitioning Single-Dimensional Data Hash partitioning can be used to distribute tuples across a set of disks. When using a relational key as the hash attribute, this approach ensures load balance, and minimal meta-data (just the hash function). However, hashing destroys data locality, and range queries are very expensive.

2 HOW RECOGNIZE IMAGE?

2.1 Detected Edge

Region boundaries and edges are closely related, since there is often a sharp adjustment in intensity at the region boundaries. The edges identified by edge detection are often disconnected. To segment an object from an image however, one needs closed region boundaries. The desired edges are the boundaries between such objects. Segmentation methods can also be applied to edges obtained from edge detectors. developed an integrated method that segments edges into straight and curved edge segments for parts-based object recognition, based on a minimum description length (MDL) criterion that was optimized by a split-and-merge-like method with candidate breakpoints obtained from complementary junction cues to obtain more likely points at which to consider partitions into different segments.

2.2 Recognized Object

When parse or segment parts at regions of deep concavity and describe those parts with common, simple volumetric terms, such as "a block," "a cylinder" "a funnel or truncated cone, We readily conduct the same process for any object, familiar or unfamiliar, in our foveal field of view. The manner of segmentation and analysis into components does not appear to depend on our familiarity with the particular object being identified.

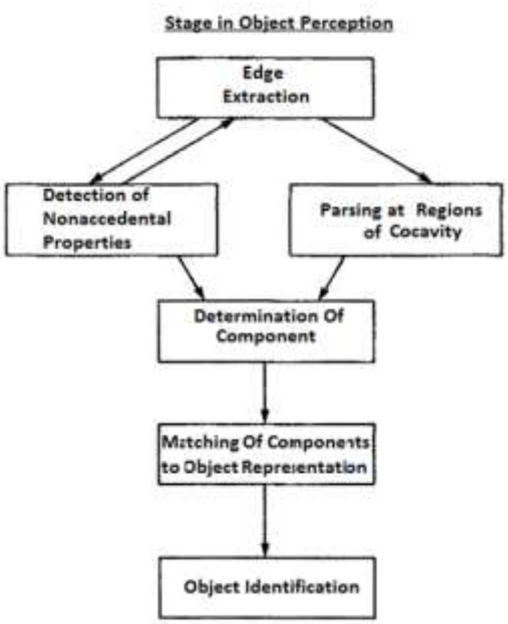

Figure 2.Presume processing stage in object recognition

2.2.1 Analysis processing stage

Edge extraction stage, responsive to differences in surface characteristics namely, luminance, texture, or color, provides a line drawing description of the object.

From this description, nonaccidental properties of image edges (e.g., collinearity, symmetry) are detected. Parsing is performed, primarily at concave regions, simultaneously with a detection of nonaccidental properties. The nonaccidental properties of the parsed regions provide critical constraints on the identity of the components. Within the temporal and contextual constraints of primal access, the stages up to and including the identification of components are assumed to be bottom-up.

The determination of an object's components should have a direct effect on the identification latency of the object. The arrangement of the components is then matched against a representation in memory. It is assumed that the matching of the components occurs in parallel, with unlimited capacity.

3 IMAGE CATEGORY

After Recognize our image then we want to recognize our object by matching with simple database that contain the simple form of shape when found comparative between our image in contain database can to classify our image and be clearly to recognize them, with this technique be easy to store and lookup ours data images in ours system.

3.1 How determine pictures classification?

After determine the category our image we can classification it from our knowledge as table1:

<table>
<tr><td rowspan="8">Manmade</td><td colspan="3">Buildings</td></tr>
<tr><td colspan="3">Bridges</td></tr>
<tr><td colspan="3">Misc</td></tr>
<tr><td rowspan="5">Transport</td><td colspan="2">Aircraft</td></tr>
<tr><td colspan="2">Ships</td></tr>
<tr><td rowspan="3">Land</td><td>Cars</td></tr>
<tr><td>Buses</td></tr>
<tr><td>Bikes</td></tr>
</table>

Table1: part of images classify

When matching our image with pattern image that exist in our system can specify exactly the classification of image by using image processing techniques. Resource description is constructed by sampling database contents via the normal process of running queries and retrieving images such as query-based sampling [6] that used for selection resources from the system when submitting query to return several different perspective images.

We use approach term frequency-inverse document frequency (tf.idf) for classification image in the same node when our data distributed in the nodes randomly without control where every image type store in the system node. Tf-idf used images in the system for determine every image in each node by percentages to classification our image correctly.

3.2 Merging Approaches

Many result-merging algorithms have been proposed in distributed information retrieval. Various approaches can be divided into two categories: approaches based on normalizing resource-specific image scores into resource-independent image scores, and approaches based on recalculating image scores at the directory service.

The Semi-Supervised Learning result-merging algorithm uses the image obtained by query-based sampling as training data to learn score normalizing functions on a query-by-query basis. It is shown to work well with a variety of resource selection and image retrieval algorithms and is the current state-of-the-art for result merging in distributed information retrieval [7]; this used the algorithm merge data from node to return the scores image.

4 LOOKUP IN P2P SYSTEM

we maps a key onto node, keyword identify by image type (classification and category) , for node identifier hash of IP address, in this way both keys and nodes are assigned an *m*-bit identifier this is a simple and common approach to use the words "nodes" and "keys" to refer to these identifiers.

If each node knows only the location of its successor, a linear search over the network could locate a particular key. This is a naive method for searching the network, since any given message could potentially have to be relayed through most of the network.
After recognize of type of images when tf.idf specified in every node in the system to determine each image type in the system, when the list of tf.idf exist at each node in the finger table the number of nodes that must be contacted to find a successor in this way can be more flexible to lookup in system when use chord lookup.

5 SIMULATION

In our simulate P2P network generation the node of DHT by Python and our query exist as database using SQL on the top distributed the database of images tf.idf stored of measure in our simulation. On each attempt sequential of finger table in node in the network from our query generator.
The studied parameters are: number of user-system interactions required to find data, interactions required to find data when user send a query and the list of result are appear of specific queries. The user selects one of image then process it to determine type of image then began search in the network. However, that the number of iterations is expected to increase when the user initiates the search with a generic query.

6 CONCLUSION

In works discover specific image for lookup in the system. We index data store for lookup the data in DHT when discover matching query, we effectiveness index in P2P images database, our data support non-exact-match must discover our type image for began lookups: one needs to know the exact key of a data item to locate the node responsible for storing that item but they depend on the exact matching facilities of the underlying DHT.

REFERENCES

A. Crespo and H. Garcia-Molina.(2002) , Routing indices for peer-to-peer systems. In Proceedings of the 22nd International Conference on Distributed Computing Systems (ICDCS),Vi-enna, Austria.

A.Rowstronand P. Druschel, (2001) Pastry: Scalable, decentralized object location, and routing for large-scale peer-to-peer systems, ” in Proceedings of the 18th IFIP/ACM International Conference on Distributed System Platforms(Middleware), (Heidelberg, Germany), pp. 329– 350.

A.Silberschatz,H.F.Korth,andS.Sudarshan,(1997) ,"Database System Concepts", chapter17.McGraw-Hill.

B.Y. Zhao, J. D.Kubiatowicz, and A.D.Joseph, (2001) Tapestry: An infrastructure for fault-tolerance wide-are a location and routing,Tech.Rep.UCB/CSD-01-1141,ComputerScienceDivision, U.C.Berkeley.

Gnutella webpage: http://gnutella.wego.com/.

J. Callan. (2000). Distributed information retrieval. In W.B. Croft, editor, Advances in Information Retrieval.Kluwer Academic Publishers. (pp. 127-150).

L. Si and J. Callan (2003). A semi-supervised learning method to merge search engine results. ACM Transactions on Information Systems, 24(4), pages 457-491. ACM.

Q. Lv, P. Cao, E. Cohen, K. Li, and S. Shenker,(2002), Search and replication in unstructured peer-to-peer networks. In Proceedings of the 16th international conference on Supercomputing .

S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker.(2001) A scalable content-addressable network. In Proceedings of ACM SIGCOMM.

Stoica, Ion et al. (2001). "Chord: A Scalable Peer-to-peer Lookup Service for Internet Applications". Proceedings of SIGCOMM'01 (ACM Press New York, NY, USA).